Using ADALM-Pluto for Monotone and Audio Signal Transmission/Reception (Normal AM)

[1]Mohammad Shamasneh, [2]Khaled Abu Lebdeh, [3]Laith Ghonem

Faculty of Engineering: Electrical and Computer Engineering Department, Birziet University
Palestine

[1]1220092-section 4

[2]1220187-section 4

[3]1221123-section 4

***Abstract*— In this project, we will design, tune, and analyze a communication system capable of transmitting and receiving monotone and audio signals using a Software-Defined Radio (SDR) platform with the Normal Amplitude Modulation-Full Carrier (AM) method. The system will be tested in real-world scenarios using the ADALM-Pluto SDR to provide practical insights into the process of signal transmission and reception. Additionally, this project aims to deepen the understanding of how signals are transmitted and received using the Normal AM-Full Carrier technique, bridging theoretical concepts with practical applications.**

***Keywords*— Normal Amplitude Modulation-Full Carrier (AM); Software-Defined Radio (SDR); ADALM-Pluto; signal transmission; signal reception; monotone signals; audio signals; communication system design.**

I. Introduction

Communication is the exchange of information between a transmitter and a receiver. It involves a systematic process where signals or messages are sent from a source to a destination through a channel. The communication system provides a structured model to describe this process, ensuring that the information is transmitted efficiently and accurately.

In any communication system, signals undergo various stages of processing. This begins with signal representation, shaping, and encoding, followed by modulation to prepare the signal for transmission. Once the signal is ready, it is passed through the transmission medium or channel. However, during its journey, the signal may encounter impairments such as noise, attenuation, and distortion, which can degrade the quality of the received information.

Effective communication systems aim to mitigate these challenges, ensuring the transmitted message is successfully received and understood at the destination. By analyzing and optimizing the system, engineers enhance the reliability and performance of the communication process. [1]

II. Amplitude Modulation (AM)

Amplitude Modulation (AM) is a basic yet significant technique in communication systems, particularly for transmitting audio signals over radio waves. It works by varying the amplitude of a carrier wave based on the message signal, allowing effective long-distance transmission. While AM is simple and historically vital in telecommunication, it has notable drawbacks, such as low power efficiency, with only 33.3% of the transmitted power carrying the message signal. Additionally, AM is highly susceptible to noise and interference, which can degrade signal quality. Despite these limitations, AM remains relevant in applications like radio broadcasting and aviation due to its straightforward implementation .[2]

III. Modulation and Demodulation in AM

In Amplitude Modulation (AM), the carrier wave c(t)c(t)c(t) serves as the foundation for transmitting the message signal over long distances. The carrier wave is expressed mathematically as: **$C(t)=A_c \cos(2\pi f_c t)$.**

In AM, the amplitude of the carrier wave is varied in proportion to the instantaneous amplitude of the message signal. This results in a modulated signal S(t), which can be described as: **$S(t) = A_c\,[\,1+K_a M(t)]\,\cos(2\pi f_c t)$.**

This equation demonstrates that the carrier wave's amplitude varies according to the message signal, allowing the transmission of information. [3]

The waveform of the modulated signal, considering both the message signal and the carrier signal as sine waves, is shown in Figure 1.

Figure 1: Waveform for m(t) and c(t) and S(t)

IV. Modulation Index

The **modulation index** (μ) in Amplitude Modulation (AM) measures how much the carrier wave's amplitude is varied by the message signal. It is defined as: $\mu = Ka \cdot Am$

Where Am is the amplitude of the message signal, and Ka is the amplitude sensitivity. The modulation index determines the depth of modulation and impacts the signal's efficiency and quality.

- **Under-Modulation ($\mu < 1$)**: Weak signal, no distortion, can received.
- **Normal Modulation ($\mu = 1$)**: Optimal transmission, can received, highest performance %33.3.
- **Over-Modulation ($\mu > 1$)**: Distorted signal due to overlapping sidebands, can't receive.

Controlling the modulation index ensures efficient transmission and prevents signal distortion. [2]

The waveform of the modulated signal based on the modulation index as we shown in Figure 2.

Figure 2: types of modulation based on the modulation index (μ)

Demodulation retrieves the original message signal from the modulated carrier wave. In AM, this process typically involves:

- **Rectification:** The carrier wave is rectified (using a envelope detector) to obtain its positive envelope, which corresponds to the message signal.
- **Filtering**: A low-pass filter removes high-frequency components, leaving only the original message signal.

This process effectively reconstructs the message signal from the modulated waveform, enabling accurate communication. [3]

V. System Design

Task 1.1:

In Task 1, we will implement a Normal AM modulation signal transmission system that transmits a signal to the receiver.:

Figure 3: Normal AM Modulation system.

We designed this system using the equation for Normal AM: $S(t)=Ac[1+KaM(t)]\cos(2\pi fct)$. First, we added the audio source to load the file source M(t). Then, we used the resampler to change the sampling rate of the audio. Next, we multiplied the modulation index Ka by the message signal. After that, we added a constant value of 1. Finally, we introduced the carrier signal and multiplied it by the modified message signal. The output signal was visualized using the QT GUI Sink to display the final result in the GUI.

Audio Source :

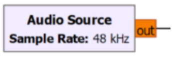

Figure 4: Audio source **Block**

Purpose: Generates audio signals at a sample rate of 48 kHz.
Output : provide an audio signal as input to the system**.**

Rational Resampler :

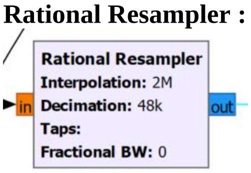

Figure 5: Rational Resampler Block

Purpose: resampling the audio signal, it's changing the sampling rate of audio.
Interpolation: The input will be 48K.
Decimation: The output will be (2M).
Output: a smapled signal incresed by 2M/48K.

Multiply Const :

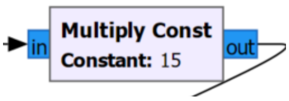

Figure 6: Multiply Const Block

Purpose: Multiplies the signal by a constant (15).
Output: A scaled signal passed to the Add Const block**.**

Add Constant:

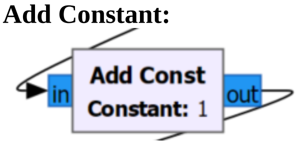

Figure 7: Add Const Block

Purpose: Addes Constant [1] to the signal.
Output: The modified signal is sent to the Multiply block.

Signal Source:

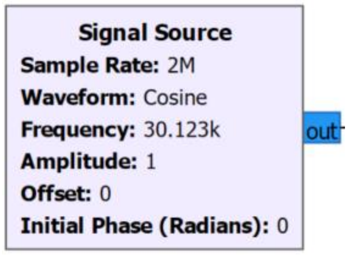

Figure 8: Signal source Block

Purpose: Generate a secondary signal[carrier signal].
Sample rate : 2M
Type : cosine
Frequancy [Fm]: 30.123K
Amplitude [Am] : 1
Output: A signal sent to the Multiply block.

Multiply :

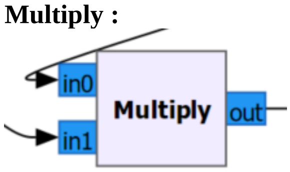

Figure 9: Multiply Block

Purpose: Multiplay the two signals to connected them, to creating a modulated signal..
Output: modulated signal sends to the QT GUI Sink.

QT GUI Sink:

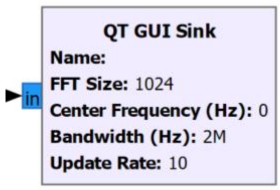

Figure 10: QT GUI Sink Block

Purpose: Visualizes the final output signal in the GUI.
Output: Displays the frequency-domain representation of the signal.

Task 1.4:

In Task 1.4, we will implement a Normal AM demodulation signal receiver that receives a message and processes it to retrieve the original signal:

Figure 13: Normal AM demodulation system.

We designed the demodulation system. First, we used the PlutoSDR Source to receive the signal from the PlutoSDR device. Then, we used the Complex to Mag [Envelope Detecter] block to convert the complex signal to magnitude. After that, the Rational Resampler block resamples the signal to match the sample rate. Finally, we added the Audio Sink to output the sound from the device.

PlutoSDR Source :

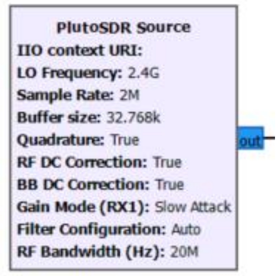

Figure 42: PlutoSDR source Block

Purpose: Receives the signal from the PlutoSDR device.
Output : Confused message to the Complex to Mag block.

Complex to Mag :

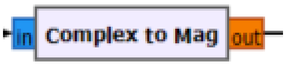

Figure 13: Complex to Mag Block

Purpose: Converts the complex signal from the PlutoSDR source into its magnitude.
Output: The output is a real-valued signal representing the amplitude of the received signal at each point in time.

Rational Resampler :

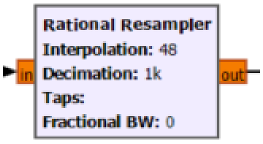

Figure 54: Rational Resampler Block

Purpose: Resamples the signal to match the target sampling rate[msg sample rate].
Output: Ensures the processed signal has the correct sampling rate for audio playback.

Audio Sink:

Figure 65: Audio Sink Block

Purpose: Outputs the processed signal as audio.
Output : The msg comes out as an audio.

Task 2 :

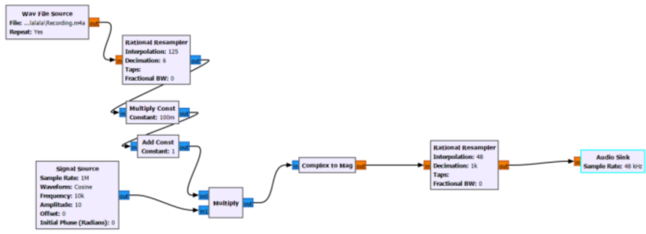

Figure 76: modulation and demodulation [Taks 2]

In Task 2, we designed a full system [modulation and demodulation]. First, we used the Wave File Source block to add the file source [msg], then sent the msg through the modulation to the demodulation to recover the audio, and the audio comes out when running the program.

Wav File Source :

Figure 87: wav File source Block

Purpose: used to load an audio file[msg].
Output : outputs the audio data from the specified file

Task 3:

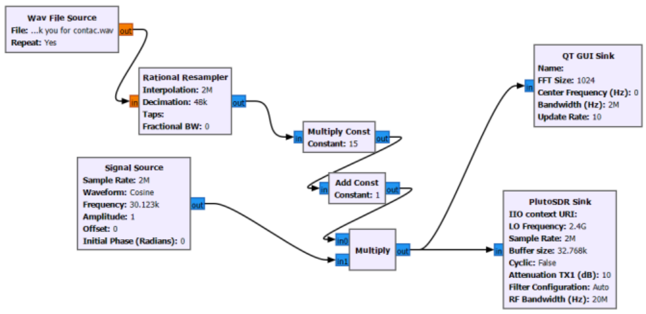

Figure 98: Normal AM modulation System with numbers configure

In Task 3.2, we implemented a demodulation process using a Low Pass Filter with a transition width of 400 Hz and a cutoff frequency of 3.5 kHz. This configuration was designed to recover only the desired message signal by filtering out unwanted high-frequency components and noise, ensuring a close demodulated output.

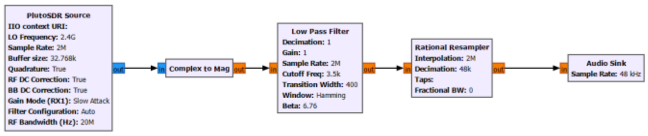

Figure 109: Normal AM demodulation using low pass Filter.

In task 3.3, we aim to optimize transmit power and receiver gain for clear signal reception. Since the message signal's frequency (Fm) is not known, we selected a carrier frequency (Fc) of 30 kHz, and set the Amplitued to 1, which proved to be a suitable values for effectively carrying the message signal.

VI. Results and Discussion

The same design of modulator and demodulator was used with different inputs. First, we used cosine wave as an input (in task 1), then we replaced it with an audio sink to modulate inputs gotten from device's microphone (in task 2). For simplicity, we will use **a square signal** as in input since that the correctness of the message and the modulated signal can be checked easily.

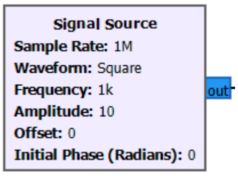

Figure 20: square signal as an input

For correct designs, the output of the demodulator must be the same (or at least similar to, if there is a distortion) to the input signal. The time domain will be used to check results clearly. We connected the input signal to the output to capture it in frequency domain, then to use to check results later.

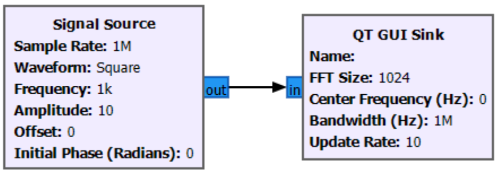

Figure 21: input signal connection

Then the following figure illustrates the input signal in time domain, to use it later for checking.

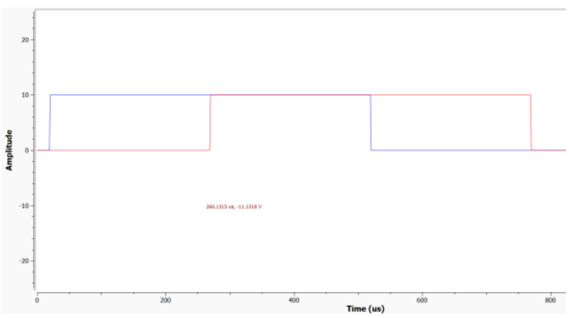

Figure 22: The message signal in time domain

The modulator implements the equation of AM modulation with appropriate value of (Ka) which results in **(μ = 1)** for better performance. After applying the message signal to the modulator specified in design part, we get the following modulated signal -in time domain-:

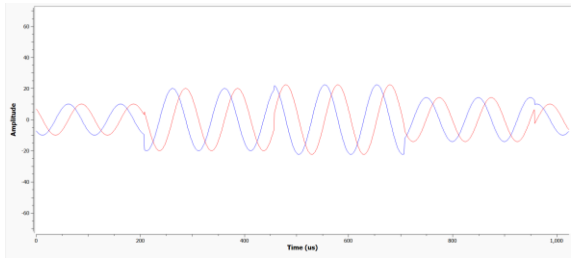

Figure 23: The modulated signal in time domain

The modulated signal is transmitted for which the demodulator to receive it and demodulate to extract the signal message from it.

The demodulator receives the signal shown in the figure above, and applies it to the demodulator scheme shown in the design part, the result of the demodulation should be similar to the original message signal.

The following figure illustrates the modulated message signal:

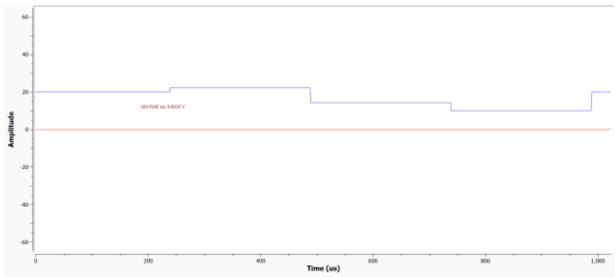

Figure 24: The demodulated signal in time domain

The demodulated signal illustrates the squareness, so the message was transmitted and received successfully.

Even though that the modulated signal is similar to the original message signal, but they are not exactly the same. In normal AM, the best performance we can get is when **(μ = 1),** and the power within the modulated signal that is concerned with the message signal is 33% from the original message. We can conclude that much power is consumed by the carrier.

The issue discussed above can illustrate the low efficiency of Normal AM modulation, which was a motivation to move towards other modulation techniques.

When we applied an **audio** signal as an input, and transmitted is using (***ADALM-Pluto***), the received message wasn't much clear. We tried to edit some configurations on blocks to have better clear audio. We chose a frequency carrier to be ten times larger than the message's frequency, and edited the value of (Ka) to let **μ** near **1.** We got better clearness in audio, but wasn't that much**.**

There are many modulation schemes that give better performance, such that Double-sided Band Supressed Carier (DSB-SC) and Single-Side Band Suppressed Carrier (SSB-SC), or even to use Angle Modulation (FM, PM).

VII. CONCLUSIONS

In conclusion, this project successfully demonstrated the design and implementation of an Amplitude Modulation (AM) system using GNU Radio and the ADALM-Pluto SDR. Through hands-on experimentation, we achieved the transmission and reception of monotone and audio signals, enhancing our practical understanding of communication systems. Key concepts such as modulation, demodulation, and parameter tuning were explored, enabling us to optimize the system for better performance and reduced noise. This comprehensive approach provided valuable insights into the practical applications of AM and the challenges associated with real-world signal transmission. Overall, the project significantly deepened our knowledge of modern communication technologies and their practical implementations.

REFERENCES

[1] Vedantu , communication system , link click here
[2] Geeksforgeeks , Amplitude Modulation , link click here
[3] Digilent , Modulation and Demodulation in AM , link click here